

## Design and Implementation of a Wireless IoT Healthcare System Based on OCF IoTivity

Wenquan Jin<sup>1</sup>, Yong-Geun Hong<sup>2</sup> and Do-Hyeun Kim<sup>3\*</sup>

<sup>1,3</sup>*Dep. of Computer Engineering, Jeju National University, Republic of Korea*

<sup>2</sup>*Electronics and Telecommunications Research Institute, Republic of Korea*

<sup>1</sup>*wenquan.jin@jejunu.ac.kr*, <sup>2</sup>*yghong@etri.re.kr*, <sup>3</sup>*kimdh@jejunu.ac.kr*

### Abstract

*The development of Internet of Things (IoT) technologies have enabled healthcare systems to use remote, autonomous and intelligent services through electronic devices such as IoT devices, smart phones and e-health sensors. The e-health is a part of the healthcare system that comprised of electronic devices with information and communication technology (ICT) for healthcare services. Open Connectivity Foundation (OCF) IoTivity is a standard solution for communications between IoT devices. In this paper, we propose a wireless IoT healthcare architecture based on OCF IoTivity. Also, we design and implement wireless e-health device, mobile e-health client, and wireless e-health server based on proposed architecture for the standard IoT communications between them.*

**Keywords:** OCF; IoTivity; Healthcare; E-health; IoT

### 1. Introduction

A healthcare system can include a management system for health elements which are used for the improvement of healthcare services for diagnosis, prevention and analysis of disease, injury and other impairments in human beings. Many countries and districts have deployed healthcare systems to manage health information. The healthcare services are one of mostly growing industry in the world. The healthcare services is also important topic because the industry is serve for human being. The people consume the healthcare service ubiquitously in the world, and with the growing population the services also need to update the quality and quantity. The World Health Organization (WHO) reported regarding to the aging problem [1]. According to the report, the problem increase social cost, and also carries new challenges to the current healthcare system. Not only aging is the issue in modern healthcare systems, but the monitoring of the health conditions for the people also needs to be improved [3]. The advanced healthcare system can reduce the cost and also works efficiently. The healthcare system can depend on the Internet and Communication Technology (ICT) to provide services using electronic devices.

Recently research in IoT technology-based healthcare is one of most concerned topic which involves sensor networks, wearables, information security and privacy [4-5]. Additionally, IoT based system consists of sensing, networking, data processing and application [6]. Therefore, a healthcare system should be a part of the internet using IoT devices and applications to improve the quality and efficiency of healthcare services. IoT healthcare systems make use of ICT for healthcare that includes IoT technologies. IoT is integration of information, communication, computation, sensing and artificial intelligence application technologies. With the IoT technologies, the IoT healthcare system can easily be operated by users such as doctors, nurses, patient, administrators, etc.

---

Received (January 8, 2018), Review Result (March 9, 2018), Accepted (March 12, 2018)

In order to develop heterogeneous applications for IoT objects, standardization is required on many layers. Several standards and frameworks are developed to build IoT healthcare systems. For example, WHO operates projects to support the strategy of e-health (E-health) applications [7]. Health Level 7 (HL 7) presented specifications for data transferring standard for healthcare systems using electronic devices [8]. Through the protocols and APIs with heterogeneous frameworks, the wireless E-health devices are able to interact with each other to exchange data. The communication between server, devices and mobile client through internet is important. The OCF has published an IoT specifications and IoTivity is frameworks for the OCF specifications [9]. IoTivity provides a communication functionalities based on Constrained Application Protocol (CoAP) [10].

In this paper, we propose the wireless IoT healthcare system that uses OCF IoTivity to implement communication between wireless E-health devices and a wireless E-health server. In the implementation, we have used Intel Edison boards to implement IoT devices. In order to develop the wireless IoT healthcare system, a design of the overall structure for the system scenario is described. Moreover, detail IoTivity messages are also described for communication in the wireless IoT healthcare system. We also have implemented and tested the proposed wireless IoT healthcare system using IoT devices with E-health sensors. The designs and implementations of the wireless IoT healthcare system are presented in details.

Rest of this paper is structured as follows; Section 2 reviews existing solutions of healthcare systems and describe OCF IoTivity. Section 3 describes the design of our proposed healthcare system. Section 4 illustrates the implementation of the wireless IoT healthcare system. Finally, we conclude our paper in Section 5.

## 2. Related Work

With advances in communication and autonomous technology, several frameworks are developed based on heterogeneous technologies to build IoT systems. Those IoT technologies can be used to develop the IoT healthcare system to interact with professionals and patients. Recently, several healthcare systems are integrated with traditional solutions in health center such as hospitals. Web services can be a great solution to expose the healthcare services on the internet for interacting with users in a ubiquitous way. In this paper, we focus on an IoT communication technology in a healthcare system which is used for communication between E-health objects.

The C4I, Common Object Request Broker Architecture (CORBA), Distributed Component Object Model (DCOM), and The Java Remote Method Invocation (Java RMI) are the topmost standards available to provide a real-time communication service [12]. These standards can be used for communication services in healthcare systems using server/client model. A real-time communication system can be more efficient with a Data Distribution Service (DDS) middleware. For a healthcare system, the middleware solution is better than CORBA, DCOM, and RMI [13]. Most of the healthcare system solutions are integrated based on web service technologies such as client/server communication paradigms. A novel approach for medical devices integration using web services is presented in [14], that illustrates the devices in the operating room is equipped with network functions. Therefore, device profiles of devices can be used as information in a network. In this proposal, a server is used as a middleware among E-health devices. The middleware receives messages from E-health devices using the HL 7 standard.

Microsoft also presented a web services based healthcare solution that is called HealthVault [15]. HealthVault provides a one-stop shop for saving the health information. The patients do not need to contact their health professionals to collect their health data histories. Those information are saved on the Microsoft Web Application where they can access their records anywhere and anytime. Heterogeneous devices provide those data and

Microsoft integrates all those information sources and provides through internet which is transparent for the patients.

In order to solve variety problem for connectivity of devices in IoT, a standard framework is needed. Heterogeneous devices are should be connected to communicate with each other such as phones, sensors and IoT devices. The Open Connectivity Foundation (OCF) has published a specification and sponsored the IoTivity open source project [16]. The OCF protocol is a RESTful interface based on HTTP/CoAP. But it has a higher level of abstraction, so multiple network communications such as bluetooth, BLE and ZigBee can be added on the protocol stack. The OCF provides mapping primitives to allow mapping of logical CRUDN commands to data communication protocol. OCF IoTivity is an open source project for implementation of the OCF specification. The IoTivity framework APIs expose the framework to developers and are available in several languages and for multiple operating systems [9]. The framework is available for Linux, Android and some specific OS such as Intel Yocto Linux and Tizen. The IoTivity works as a platform to allow multiple protocols for data transmission between OCF server and OCF client. Also it supports several functionalities, i.e. management, registration, discovery, monitoring, and maintenance functions. The IoTivity can be used for communication between E-health devices as a standard solution.

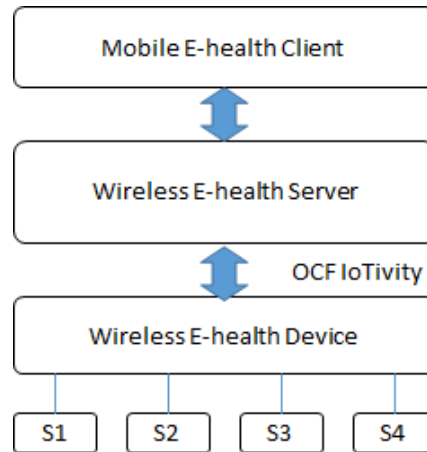
### **3. Design of IoT Healthcare System based on OCF IoTivity**

#### **3.1. IoT Healthcare Architecture**

IoT technology is widely applied for all area, particularly in healthcare with an objective to provide efficient services through mobile devices and also support healthcare services for remote areas [17]. Sensing and remote monitoring improve the quality of healthcare services and help in reducing the healthcare cost [18]. Wearable devices and E-health sensors can be part of wireless sensor networks through communication technology for healthcare system [19]. In this section, the wireless IoT healthcare system is introduced.

Figure 1 shows the proposed architecture of wireless IoT healthcare system based on OCF IoTivity. The architecture includes three components for healthcare services as shown in the figure. The wireless E-health device is an end user device that can support sensing, monitoring and displaying functions. The device is an IoT device that also need to work in limited power supply environment for long time. The device can equip multiple sensors to get the health information and equip screen to display the health information to the professionals. The health information is stored in the wireless E-health server that includes the database for storing information of users and health, and others. The mobile E-health client is used for interacting with the wireless E-health server and wireless E-health devices. Users can consume services to control wireless E-health devices through mobile E-health clients and that are provided by the wireless E-health server. Between the wireless E-health server and wireless E-health devices, we suggest the OCF IoTivity to implement the interworking mechanism for the wireless IoT healthcare system.

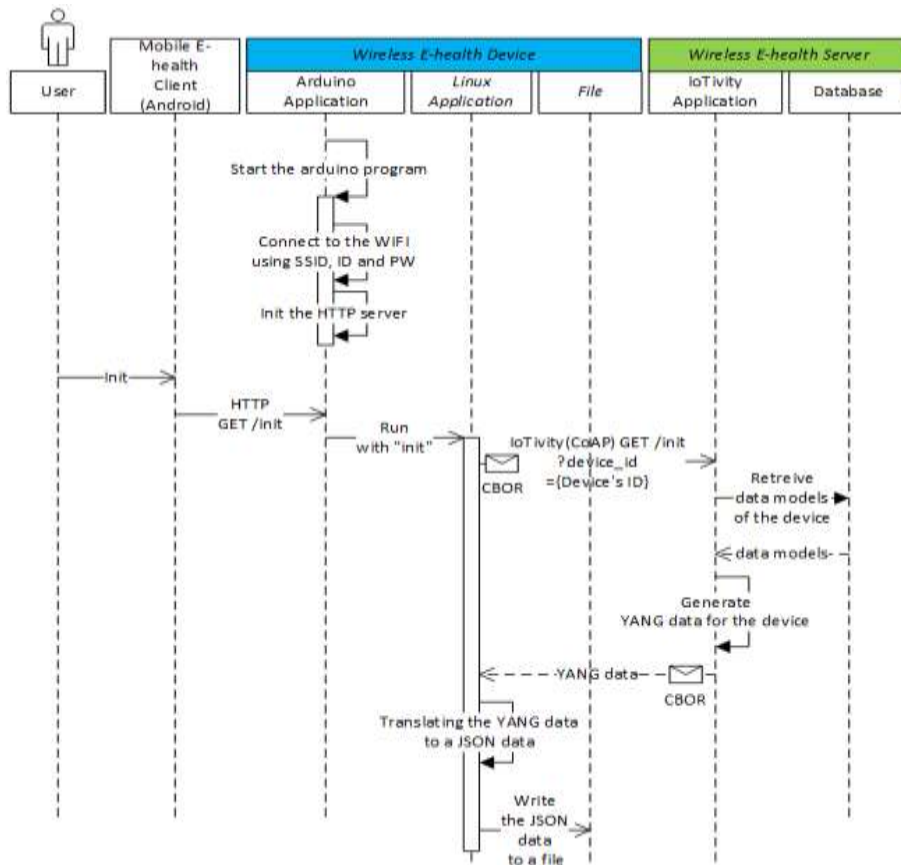
A healthcare system may include users, wireless E-health devices, client and wireless E-health server. Users can be professionals and patients. A patient can consume healthcare services through client that can control wireless E-health devices. The devices can be E-health sensors which are used for getting health data from the patient. The health data are needed to send to wireless E-health server in order to be used by users. In this part, IoTivity is used for the healthcare system. Therefore, IoTivity client is deployed in the wireless E-health device side and IoTivity server deployed in wireless E-health server side. In order to save health data as well as other data, wireless E-health server has a database.



**Figure 1. Proposed Architecture of the Wireless IoT Healthcare System Based on OCF IoTivity**

### 3.2. IoT Healthcare Sequence Diagram

Figure 2 shows the OCF IoTivity based communication sequence for requesting configuration data from the wireless E-health server.



**Figure 2. OCF IoTivity based Communication Sequence for Requesting Configuration Data from Server**

Before the process, the wireless E-health device needs to connect WIFI and initialize the HTTP server for controlling. After the user of the device confirms the process, the user is able to initialize the device configuration. The configuration is

based on a semantic data which is saved in a wireless E-health server in the system. The user uses a client to control the wireless E-health device to request the semantic data from the wireless E-health server. As shown in the sequence diagram, when the wireless e-health device's Arduino application receives the HTTP GET request, the application runs a Linux application in the device. The application sends IoTivity request to the wireless E-health server with the device's ID. The semantic data which is a YANG data. The YANG data is received by the device, the device translates the YANG data to a JSON data and writes to a file to save the semantic data. So far is the process for initializing the configuration.

### 3.3. JSON to IoTivity in IoT Healthcare

We have used the three basic kinds of message structures *i.e.*, Leaf, container and list for health data message structure. The original meaning of those categories is defined in YANG specification. The IoTivity messages of health data in the system are generated by JSON configuration data. It means when a wireless E-health device generates an IoTivity message then the wireless E-health device refers the JSON configuration data.

Figure 3 shows a part of JSON configuration data and its IoTivity payload. A body temperature sensor collects health data from a patient and generates a single value in the system. In this case, leaf category is fitted. In the IoTivity payload, the value has a single value.

JSON Configuration Data	IoTivity Payload
<pre>{   "name": "BodyTemperatureURI",   "description": "Body Temperature Sensor",   "value": {     "prefix": "leaf",     "type": "string"   } }</pre>	<pre>{   "unit-id": "BodyTemperatureURI",   "time": "20160910145418",   "value": "33.10" }</pre>

**Figure 3. JSON Configuration Data and IoTivity Payload for Leaf**

The container is used for multiple sensing data. For example, a sensor collects 3 type of sensing data in a moment. The "container" statement is the value of "prefix" in JSON. The value of "type" in JSON that multiple objects are included in for multiple data. Through this structure, the IoTivity payload is generated as shown in Figure 4. The value in the IoTivity payload which includes 4 kinds of values and defined in the JSON configuration data.

JSON Configuration Data	IoTivity Payload
<pre>{   "name": "BloodPressureResURI",   "description": "Blood Pressure Sensor",   "value": {     "prefix": "container",     "type": {       "time-data": "string",       "systolic-data": "string",       "diastolic-data": "string",       "pulse-data": "string"     }   } }</pre>	<pre>{   "unit-id": "BloodPressureResURI",   "time": "20160910145427",   "value": {     "time-data": "201609101433",     "systolic-data": "132",     "diastolic-data": "80",     "pulse-data": "96"   } }</pre>

**Figure 4. JSON Configuration Data and IoTivity Payload for Container**

Figure 5 shows a part of JSON configuration data and its IoTivity payload for list category. The JSON configuration data is used for generating a message which carries a sequential sensing data. Through this structure, the IoTivity payload is generated as shown in the figure. The value in the IoTivity payload which includes an array of values which defined in the JSON configuration data. An electromyography sensor collects health data from a patient continuous. Therefore, the list statement is used for the case.

JSON Configuration Data	IoTivity Payload
<pre>{   "name": "EMGResURI",   "description": "Electromyography Sensor",   "value": {     "prefix": "list",     "type": "string"   } }</pre>	<pre>{   "unit-id": "EMGResURI", "time": "20160910145434", "value": [     "330.00", "390.00", "411.00", "381.00", "399.00"   ] }</pre>

**Figure 5. JSON Configuration Data and IoTivity Payload for List**

JSON Configuration Data	IoTivity Payload
<pre>{   "name": "GSRResURI",   "description": "Galvanic Skin Response Sensor",   "value": {     "prefix": "list",     "type": {       "conductance-data": "string",       "resistance-data": "string",       "conductancev-data": "string"     }   } }</pre>	<pre>{   "unit-id": "GSRResURI", "time": "20160910145450", "value": [     {       "conductance-data": "-1.000000", "resistance-data": "-1.000000", "conductancev-data": "0.493646"     },     {       "conductance-data": "-1.000000", "resistance-data": "-1.000000", "conductancev-data": "0.498534"     },     {       "conductance-data": "-1.000000", "resistance-data": "-1.000000", "conductancev-data": "0.498534"     },     {       "conductance-data": "-1.000000", "resistance-data": "-1.000000", "conductancev-data": "14614300.000000"     },     {       "conductance-data": "-1.000000", "resistance-data": "-1.000000", "conductancev-data": "0.493646"     }   ] }</pre>

**Figure 6. JSON Configuration Data and IoTivity Payload for List for Multiple Sensing Output**

Figure 6 shows a part of JSON configuration data and its IoTivity payload for list category. But it is used for multiple sensing outputs. A GSR sensor continuously collects conductance-data, resistance-data and conductance-data. In this case, the JSON Configuration is used.

## 4. Implementation of IoT Healthcare System

### 4.1. Implementation Environment

In order to develop the proposed system, we need three different components *i.e.*, mobile E-health client, wireless E-health device and wireless E-health server.

Each is a project that built in a specific development environment. Table 1 shows the details of development for the IoT healthcare system. The information is presented for the component, development OS, device, programming language, development tool, library/framework, and database (DBMS). The implementation of the system is done by Windows 10 Pro b4 bit OS. We use the Android Phone (API Level 22) for mobile E-health client, Intel Edison Board (Intel Yocto Linux Image 0606) for wireless E-health device, and Intel Edison Board (Intel Yocto Linux Image 0606) for wireless E-health server. The programming language, we use Java for mobile E-health client, C++ and Arduino for wireless E-health device, and C++ for wireless E-health server. The

development tool, we use Android Studio 2.1 for mobile E-health client, Eclipse IDE by Intel and Arduino 1.6.1 for wireless E-health device, and Eclipse IDE by Intel for wireless E-health server. We use the californium CoAP framework to implement the CoAP communication of Java application in this system. Therefore, we use the library to implement the mobile E-health client. For HTTP communication, the Volley also is used in the client application. For wireless E-health device the IoTivity library is used for OCF communication and MRAA library is used for collecting sensing data from sensors. The E-health tool kit library and wifi library are used in the Arduino application. For the server, we use IoTivity library and libcoap library for implementing the communication functions. The database in the server, we use SQLite 3 to support the data writing and reading functions.

**Table 1. Development Environment**

Component	Mobile E-health Client	Wireless E-health Device		Wireless E-health Sever	
Development OS	Windows 10 Pro 64bit				
Device	Android Phone (API Level 22)	Intel Edison Board (Intel Yocto Linux Image 0606)		Intel Edison Board ( Intel Yocto Linux Image 0606)	
Programming Language	Java	C++	Arduino	C++	C
Development Tool	Android Studio 2.1	Eclipse IDE by Intel	Arduino 1.6.1	Eclipse IDE by Intel	
Library/Framework ork	Californium CoAP Framework, Volley	IoTivity 1.0.0, MRAA 1.1	E-health Tool Kit Library, WIFI Library	IoTivity 1.0.0	Libcoap
DBMS				SQLite 3	

#### 4.2. IoT Healthcare Sensor

We have used different sensors to test the proposed architecture. Each sensor collects different data format as shown in the Figure 3-6. The body temperature sensor collects the body temperature from a person that can measure in Celsius. Through the programming, the sensor collects 5 time continuous sensing data and calculates those data to a mean value. The blood pressure sensor collects blood pressure-related information from a person that measures 4 type of sensing value in a moments. It mean the sensor support a set of data to the user. The electromyography sensor collects muscle information that support an array data. Using the continuous data professionals can diagnose the disease. The galvanic skin response sensor collects continuous data with conductance, resistance and conductance sensing data.



**Figure 7. E-health Sensor Shield**

In this healthcare system, E-health Kit is used which is developed by Libelium. The kit works on Arduino platform for getting health data. The sensors work with Arduino application through the shield as shown in Figure 7. In order to use the sensors, the shield is needed to set on an Arduino board. In this development, we have used an Intel Edison board.

Figure 8 shows the wireless E-health device. The device consists of an Intel Edison board and E-health Kit shield. For this development we have used Libelium kit having body temperature sensor, blood pressure sensor, electromyography sensor and galvanic skin response sensor with the E-health kit shield.



**Figure 8. Wireless E-health Device**

#### **4.3. IoT Healthcare Server**

Figure 9 shows the wireless E-health server which is an Intel Edison board. The server applications are hosted in the board. The Intel Edison board support a Linux operating system with Arduino platform. The IoTivity framework is deployed in Linux operating system for supporting services in the healthcare system.



**Figure 9. Wireless E-health Server**

#### **4.4. Sensor's Information List in E-health Client**

Figure 10 shows the screenshot of client application that present a detail information of health resource and buttons for controlling the sensor and getting history data from the server. The health resource present the service that provided by the E-health device for the related sensor. The mobile E-health client is used for controlling a wireless E-health device and consuming services from the wireless E-health server. The figure shows a client which is used for controlling a wireless E-health device and consuming services from a wireless E-health server. There are 4 functions in the user interface. The first button is used for activating an E-health sensor in a wireless E-health device to collect health data from the user. The second button is used for requesting the wireless E-health device to send health data to the E-health server through IoTivity. The third button is used for requesting the user's historical health data from the wireless E-health server. The fourth button is used for requesting the user's health data from the wireless E-health device that the data exist in the device.





**Figure 10. Mobile E-health Client**

## 5. Conclusions

In this paper, we have proposed the wireless IoT healthcare architecture based on OCF IoTivity. The OCF IoTivity is a standard that is able to implement IoT devices. With the IoT technologies we have designed and implemented the wireless IoT healthcare system that involves mobile E-health client, wireless E-health device and wireless E-health server. In order to apply the IoTivity to the wireless IoT healthcare system, we developed wireless E-health device on Intel Edison board with E-health sensors which communicate with wireless E-health server through IoTivity. The wireless E-health server provides IoTivity services which are used for configuring wireless E-health devices and saving health data to the server. Moreover, for the usage of IoTivity in the wireless IoT healthcare system, we also described the performance that can be referred to other research.

## Acknowledgments

This work was supported by the National Research Council of Science & Technology (NST) grant by the Korea government (MSIT) (No. CRC-15-05-ETRI), and this work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT) (No.2017-0-00756, Development of interoperability and management technology of IoT system with heterogeneous ID mechanism). Any correspondence related to this paper should be addressed to DoHyeun Kim; kimdh@jejunu.ac.kr. This paper is a revised and expanded version of a paper entitled "A Study of Wireless IoT Healthcare Systems using OCF IoTivity" presented at CRTT 2017 [11].

## References

- [1] World Health Organization, World Health Statistics 2013, (2014).
- [2] Department of Economic and Social Affairs Population Division, World Population Ageing 2013, United Nation, New York, (2013).
- [3] E. Balandina, "IoT use cases in healthcare and tourism", Business Informatics (CBI), 2015 IEEE 17th Conference on IEEE, vol. 2, (2015).
- [4] B. PL Lo, H. Ip and G.-Z. Yang, "Transforming health care: body sensor networks, wearables, and the Internet of Things", (2016).
- [5] P. Gope and T. Hwang, "BSN-Care: A secure IoT-based modern healthcare system using body sensor network", IEEE Sensors Journal, val. 16, no. 5, (2016), pp. 1368-1376.
- [6] C. Harold Liu, B. Yang and T. Liu, "Efficient naming, addressing and profile services in Internet-of-Things sensory environments", Ad Hoc Networks, vol. 18, (2014), pp. 85-101.

- [7] Standards and eHealth, ITU-T Technology Watch Report, (2011) January.
- [8] J. José P.C. Rodrigues, S. Sendra Compte and I. de la Torra Diez, "Health Level 7", Advances on Sensors for Health Systems, (2016).
- [9] OCF IoTivity, <https://www.ietf.org/>, (2017) January.
- [10] C. Bormann and Z. Shelby, "Block-Wise Transfers in the Constrained Application Protocol (CoAP)", (2016).
- [11] W. Jin and D. H. Kim, "A Study of Wireless IoT Healthcare Systems using OCF IoTivity", The 1st International Conference on Convergence Research Theory and Technology CRTT 2017, (2017), pp. 15-16.
- [12] M. Fang. "Design and implementation of real-time distributed system using DDS", Proceedings of IEEE international conference on automatic control and artificial intelligence, (2010).
- [13] B. Almadani, B. Saeed and A. Alroubaiy, "Healthcare systems integration using Real Time Publish Subscribe (RTPS) middleware", Computers & Electrical Engineering, vol. 50, (2016), pp. 67-78.
- [14] D. Gregorczyk, T. Bußhaus and S. Fischer, "A proof of concept for medical device integration using web services", Systems, Signals and Devices (SSD), 2012 9th International Multi-Conference on IEEE, (2012).
- [15] HealthVault. <https://www.healthvault.com/>, (2017) January.
- [16] Open Connectivity Foundation. <https://openconnectivity.org/about>, (2017) January.
- [17] P. A. Laplante and N. Laplante, "The internet of things in healthcare: Potential applications and challenges", IT Professional, vol. 18, no. 3, (2016), pp. 2-4.
- [18] L. Catarinucci, "An IoT-aware architecture for smart healthcare systems", IEEE Internet of Things Journal, vol. 2, no. 6, (2015), pp. 515-526.
- [19] P. Castillejo, "Integration of wearable devices in a wireless sensor network for an E-health application", IEEE Wireless Communications, vol. 20, no. 4, (2013), pp. 38-49.

## Authors



**Wenquan Jin**, received the B.S. degree in computer science and technology from Yanbian University of Science and Technology, China, in 2013, and the M.S. degrees in computer engineering from the Jeju National University, Korea, in 2015. Now he is a Ph.D candidate in the Jeju National University, Korea. His research interests include sensor networks, M2M/IOT, IoT intelligent service, and mobile computing.



**Yong-Geun Hong**, received the B.S. degree in electronics engineering from the Kyungpook National University, Korea, in 1997, and the M.S. and Ph.D. degrees in information telecommunication the Kyungpook National University, Korea, in 1999 and 2013, respectively. Since 2001, he has been with the Electronics and Telecommunications Research Institute, Republic of Korea. His research interests include sensor networks, M2M/IOT.



**Do-Hyeun Kim**, received the B.S. degree in electronics engineering from the Kyungpook National University, Korea, in 1988, and the M.S. and Ph.D. degrees in information telecommunication the Kyungpook National University, Korea, in 1990 and 2000, respectively. He joined the Agency of Defense Development (ADD), from March 1990 to April 1995. Since 2004, he has been with the Jeju National University, Republic of Korea, where he is currently a Professor of Department of Computer Engineering. From 2008 to 2009, he has been at the Queensland University of Technology, Australia, as a visiting researcher. His research interests include sensor networks, M2M/IOT, energy optimization and prediction, intelligent service, and mobile computing.